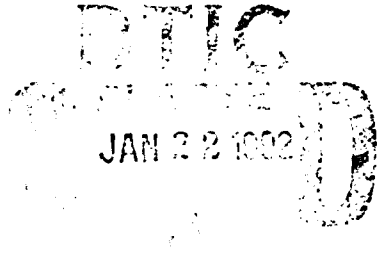
December 1991

M91-87

Jonathan K. Millen

The Cascading Problem



MITRE

Bedford, Massachusetts

December 1991

M91-87

Jonathan K. Millen

The Cascading Problem

CONTRACT SPONSOR National Computer Security Center
CONTRACT NO. DAAB07-91-C-N751
PROJECT NO. 8812Q
DEPT. G110

MITRE

The MITRE Corporation
Bedford, Massachusetts

ABSTRACT

This paper characterizes the cascading problem, discusses the assumptions and rationale behind cascading analysis, and summarizes what is known about detecting it analytically. The nesting condition and a matrix algorithm are given. Examples illustrate the insufficiency of the original nesting condition in a partially-ordered-level environment, and the danger of treating networks as single systems unless they have a uniform penetration resistance.

ACKNOWLEDGMENTS

The author is grateful for many helpful conversations, leading in some cases to specific results noted in the text, with Leonard Monk, Sam Schaen, and Todd Wittbold.

TABLE OF CONTENTS

LIST OF FIGURES

SECTION 1

INTRODUCTION

When a trusted computer system is connected to a network, there are new risks to consider, risks to the network as well as to the computer system. Each party to the connection wishes to ensure that the sensitive data it protects will continue to be adequately protected when it is exported across the connection. Obviously, the system receiving the exported information should be capable of protecting it; this is a local risk factor. It is less obvious that there are also global risks, which depend on the topology of the network. The cascading problem is one of those global risks. This paper characterizes the problem and summarizes what is known about detecting it analytically. It includes a matrix algorithm reported previously only in less accessible references, and discusses the assumptions and rationale behind cascading analysis.

Cascading is a concern in networks where dissemination of information is limited on the basis of a sensitivity label and some less standardized need-to-know controls. The cascading problem is concerned with access control based on the sensitivity label. Respecting need-to-know restrictions is an important consideration in network security policy, but it has an impact on the cascading problem only to the extent that it influences the topology of the network.

Sensitivity label restrictions in a multilevel network are handled by assigning each subsystem an *accreditation range*, defining the set of sensitivity levels that a subsystem is trusted to segregate and label accurately for export over network links. By a "subsystem," we mean a computer system or subnetwork treated as a unit for purposes of analyzing the network, and possessing an accreditation range. If a subsystem is itself a network, it should be a unified system rather than an arbitrary accredited interconnection, for reasons given at the end of Section 3.

These concepts, and the cascading problem itself, arose in work relating to DOD applications. The notion of an accreditation range comes from DOD Directive 5200.28 [1]. Appendix C of the Trusted Network Interpretation (TNI, [2]) describes the role of accreditation ranges in connecting automated information systems into a network. Appendix C of the TNI also contains the first published discussion of the cascading problem, although the concern was brought up originally by Stephen Walker and James Anderson during separate working group discussions in the 1985 DOD Workshop on Network Security [3]. The cascading problem is also summarized in the TNI Environments Guideline [4]. An example application of the TNI techniques was

described by Powanda and Genovese [5]. Fitch and Hoffman suggest that similar concerns may also apply in a commercial or public environment [6].

HOW CASCADING OCCURS

The cascading problem would not exist if a subsystem could be trusted absolutely with information within its accreditation range. However, this is not ordinarily the case. The idea of an accreditation range is that there is an *acceptable risk* in placing information at the high end of the range into a computer system that will export information labelled at the low end of the range, to users or other systems.

What happens when subsystems with different accreditation ranges are interconnected? We assume that links between systems carry only information at some level common to both accreditation ranges. However, the network may span a greater range of sensitivity levels than either subsystem, as shown in Figure 1.

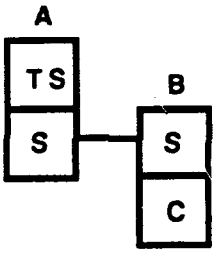


Figure 1. A Network With a Possible Cascading Problem

System *A* in Figure 1 is accredited for Secret (S) and Top Secret (TS) information, as indicated by its subdivision into labelled sections. System *B* is accredited for Secret (S) and Confidential (C) information. Secret information is conveyed back and forth between the two systems on a network link. The network as a whole now carries information from TS through C. Is the risk still acceptable? If not, there is a cascading problem.

The risk in question is the possibility that some TS information in System *A* might be downgraded to the S level through the action of malicious software; then transmitted at the S level across the network link to System *B*; and subsequently released at level C, through the action of cooperating malicious software in System *B*. Again, these risks would not have to be considered if systems *A* and *B* were invulnerable within their respective accreditation ranges. But since the risk of successful attack by malicious software exists in each system, we must consider the

possibility of both attacks working together to accomplish the compromise of TS information at the C level.

EVALUATING THE RISK

Before proceeding, we must ask how serious this risk is, and how it compares with those normally accepted in standalone systems. Note that a successful attack requires not only that both subsystems involved have flaws, but also that appropriate malicious software has been installed to exploit them, and that all instances of the malicious software cooperate actively across the network. This presents a greater obstacle to the potential penetrator than each of the individual subsystems alone. The question is whether it presents a greater obstacle than a subsystem qualified to handle the expanded range of sensitivity levels.

The risk of multiple penetrations leading to cascading is discussed by Lee [7], in the context of an interconnection of systems that have been evaluated using the DOD Trusted Computer System Evaluation Criteria (TCSEC) [9]. The TCSEC gives criteria for measuring the penetration resistance of computer systems. It assigns to each system one of the evaluation classes D, C1, C2, B1, B2, B3, or A1, in order of increasing strength. The question addressed by Lee is whether an interconnection of two systems of strength B2, for example, offers significantly greater penetration resistance than each one alone, if a penetrator must overcome both of them. Lee takes a probabilistic approach, assuming that the penetration resistance is related to the probability that some flaw in the system will be discovered.

Lee gives an argument that if two systems have probabilistically independent flaw sources, "...the resistance to threat of a cascade of two B2 systems is approximately the same as, or even better than, that of a B3 system." But Lee also remarks that demonstrating effective independence of flaw sources in a practical case is not easy. Also, two systems may have the same or equivalent flaws, particularly if their operating systems are the same, or are separate implementations of a single flawed design. Moreover, while exploitation of the flaws on two or more systems does present additional resistance to the attacker, it should be kept in mind that physical access to all interconnected systems is not necessary to send untrusted software to them, as the proliferation of viruses shows unmistakably.

It has been observed that even when two standalone systems are not connected by a network, personnel can load a tape on one system, carry it physically over to the other system and mount it, and continue processing the information just as though it had been transferred over a communications line. This has sometimes been used as an

argument that if we did not have to worry about cascading for standalone systems, there is no need to worry about it for networks. On the other hand, if the tape transfers occur routinely and often, it could just as well be an argument that a cascading risk should be considered for those systems.

RISK MEASUREMENT

The analysis to be presented here is based on the existence of a metric for risk, i.e., a *risk index*, and on worst-case assumptions about the propagation of risk over a network. A risk index is supposed to quantify the level of threats that might be mounted against a multilevel system, based on the value of the information that might be compromised. We would like risk to be measured on the same scale as penetration resistance, so that we can compare one to the other to decide whether the system is strong enough to meet the expected threats to it.

Historically, there has been an accepted approach to quantify risk, at least to a first approximation. One looks at the difference between the highest sensitivity of data on the system and the lowest level of clearance of system users. This approach began with the notion of "controlled mode" operation in the original DOD Directive 5200.28, in which the high and low were two adjacent levels. The measure was recognized as a "risk index" in the Environments Guideline or so-called "Yellow Book" [8], and reappeared in Enclosure 4 of the new DOD Directive 5200.28 [1].

Cascading analysis can be carried out with a variety of risk metrics, and the DOD version is only one of the possibilities. Certain assumptions about the risk index will be needed, however. In this paper, we assume that the risk index is a non-negative function $R(a,b)$ of pairs of sensitivity levels, such that $R(a,b) = 0$ if $a \leq b$. $R(a,b)$ represents the risk associated with a downgrade from a to b. The risk index of an accreditation range is the least upper bound of $R(a,b)$ for a and b within the range. Additional assumptions will be added below where needed for specific conclusions.

PENETRATION RESISTANCE

The *penetration resistance* of a computer system is defined here as the maximum risk index of the accreditation ranges that could be assigned to it. This terminology presumes that accreditation ranges are assigned soundly, so that the features, assurance, and administrative context of each system justify the trust that has been placed in it.

Note that the accreditation range actually assigned to a system in a particular network may be smaller (in risk index) than its penetration resistance would justify. In fact, sometimes the solution to a cascading problem is to replace a system by another system having the same accreditation range but a greater penetration resistance.

It is beyond the scope of this paper to discuss how the design of a computer system is evaluated or certified to determine its penetration resistance. We remark only that the DOD recommends a set of Trusted Computer System Evaluation Criteria [9] to determine an evaluation class. Having determined the class, the "Minimum Class" table in the Yellow Book and other DOD documents indicates the maximum risk index it should support.

To decide whether a cascading problem exists, we will also have to estimate the penetration resistance of a collection of subsystems, along a cascading path. The results in this paper are based on the worst-case or conservative assumption that the penetration resistance of a collection of systems is the maximum of their individual penetration resistances. It is pointed out in [6] that the matrix algorithm for calculating the path resistance can be adapted to certain other combining functions.

DEFINITION OF CASCADING

According to the TNI,

> "The cascading problem exists when a penetrator can take advantage of network connections to compromise information over a range of security levels that is greater than the accreditation range of any of the component systems he must defeat to do so."

A somewhat clearer definition, using the terminology introduced above, is the following:

> The cascading problem exists when a penetrator can take advantage of network connections to compromise information over a range of security levels that is greater in risk index than the penetration resistance of the collection of subsystems that must be defeated to do so.

MOTIVATION FOR CASCADING ANALYSIS

If evaluating a cascading risk requires some detailed consideration of the subsystems involved, then there is a benefit to limiting that work to those subsystems in which cascading might actually occur. Cascading is possible only in some network configurations, and then only along certain paths in the network. Suppose, for example, that Systems *A* and *B* in Figure 1 were not connected to one another. Then the cascading problem in Figure 1 would disappear. It would also disappear if either system were accredited from TS through C, since the risk of defeating that one system is already small enough to be acceptable.

Clearly, the existence and locality of cascading paths depend on the connectivity of the network, and the accreditation ranges of the subsystems in it. Analysis of this information alone can eliminate or localize possible cascading paths. This is the motivation behind having techniques such as those in Appendix C of the TNI and in this paper: an ounce of analysis can prevent a pound of risk assessment.

SECTION 2

TESTS FOR CASCADING

Various ways have been suggested to detect or prevent cascading. Walker observed that cascading cannot occur if all the accreditation ranges in the network have the same top level. Schaen (of The MITRE Corporation) proposed the "nesting condition," given in Appendix C of the TNI, summarized in [12], which is applicable when sensitivity levels are linearly ordered. The TNI also suggests a heuristic procedure and a graph-theoretic characterization. A Prolog program to detect cascading is given by Millen and Schwartz [10]. That paper also contains a generalized nesting condition that works for partially ordered sensitivity levels, found by Monk (of The MITRE Corporation). A matrix algorithm was suggested by Millen [11] and implemented in the ANSSR network risk analysis system [13]. Fitch and Hoffman present a network security model that handles cascading as a special case, using a more efficient matrix algorithm [6].

The two most useful methods are the nesting condition, because of its simplicity, and a form of the matrix algorithm, because it is the most efficient exact test; so these two will be given below. The nesting condition is pessimistic, in the sense that, if it fails, there is not necessarily a cascading problem. But, in an environment with linearly ordered sensitivity levels, it is conservative, in the sense that, if it succeeds, there is definitely no cascading problem. The matrix algorithm is exact with respect to the stated assumptions about risk measurement and penetration resistance. In a broader sense, its results may still be pessimistic, in view of the fact that the definition of cascading is based on worst-case assumptions about penetration resistance.

THE NESTING CONDITION

The Original Nesting Condition

A network satisfies the *nesting condition* as stated in Appendix C of the TNI [4] if the accreditation ranges of its subsystems are pairwise either disjoint or nested. A pair of accreditation ranges are disjoint if they have no levels in common. A pair of accreditation ranges are nested if one is a subset of the other.

The nesting condition does not take network connectivity into consideration. All possible pairs of accreditation ranges (not just those of adjacent subsystems) must be compared, but some pairs may be nested and others disjoint.

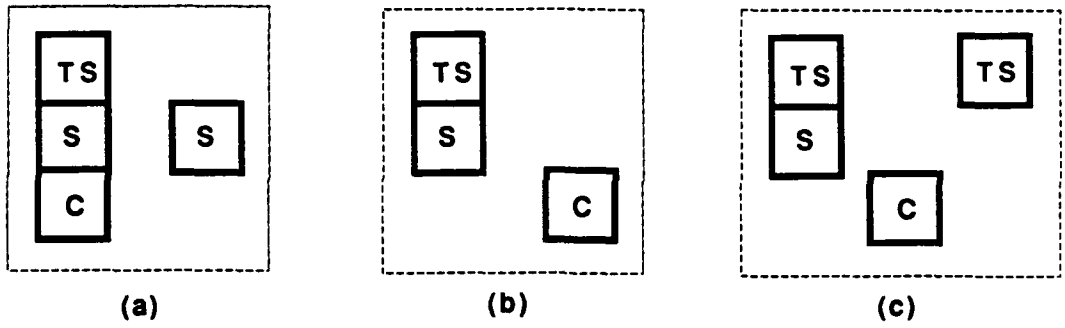


Figure 2. Nested (a), Disjoint (b) and Mixed (c) Accreditation Ranges

Figure 2 shows examples of sets of accreditation ranges that satisfy the nesting condition. Figure 3 shows a set of accreditation ranges that fails the nesting condition.

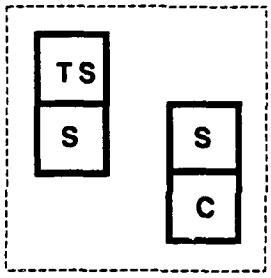


Figure 3. Accreditation Ranges Failing the Nesting Condition

The pessimism of the nesting condition is evident because the ranges in Figure 3 might have belonged to two disconnected systems, which could not have a cascading problem.

Satisfaction of the nesting condition is sufficient to prevent cascading if the set of sensitivity levels is linearly (i.e., totally) ordered. This is true, in particular, if they are pure DOD classifications, with no compartments or categories. The sufficiency of the nesting condition follows from a theorem about the generalized nesting condition given in [10]. For completeness, we state the generalized nesting condition here.

The Generalized Nesting Condition

A network satisfies the *generalized* nesting condition if the accreditation ranges of its subsystems are pairwise either nested, incomparable, or strictly ordered. A pair of accreditation ranges are *incomparable* if every element of one is incomparable (in the

partial ordering of sensitivity levels) to every element of the other. They are *strictly ordered* if every element of one is strictly dominated by every element of the other. If sensitivity levels are totally ordered, the generalized nesting condition reduces to the original nesting condition. This definition is due to Leonard Monk.

The generalized nesting condition has been proved sufficient to prevent cascading in systems where certain additional assumptions have been made about accreditation ranges and the risk index. Accreditation ranges should be convex, and the risk index should be interval-based and skew-monotonic. These concepts are defined as follows:

An accreditation range is *convex*, if whenever a, b, and c are levels in a chain $a > b > c$, such that a and c belong to an accreditation range, then the element in between, b, must also belong to that accreditation range.

A risk index $R(a,b)$ is *skew monotonic* if:

(1) $R(a,a) = 0$
(2) if $c \leq a$, then $R(c,b) \leq R(a,b)$, and
(3) if $b \leq c$, we have $R(a,c) \leq R(a,b)$.

There was a stricter form of skew monotonicity in [10] that was used only for a converse result, and it is not needed here. Note that if $a \leq b$, then by skew monotonicity $R(a,b) \leq R(b,b) = 0$, so $R(a,b) = 0$.

THEOREM (See [10]): *If*
(1) a network satisfies the generalized nesting condition,
(2) the accreditation ranges of its subsystems are convex, and
(3) the risk index is skew monotonic,
then it has no cascading problem.

When accreditation ranges are convex, we can see that the generalized nesting condition reduces to the original nesting condition if sensitivity levels are totally ordered. For, in a totally ordered system, a convex set is simply an ordered chain of elements. If two accreditation ranges are disjoint, they must be strictly ordered.

Partially Ordered Levels - A Counterexample

If sensitivity levels are partially but not totally ordered, the original nesting condition does not necessarily prevent cascading. A counterexample, not published previously, is shown in Figure 4. The sensitivity levels shown are the DOD

classification/category set pairs with the usual lattice ordering. For example, S/AB indicates a classification of Secret and the category set {A,B}.

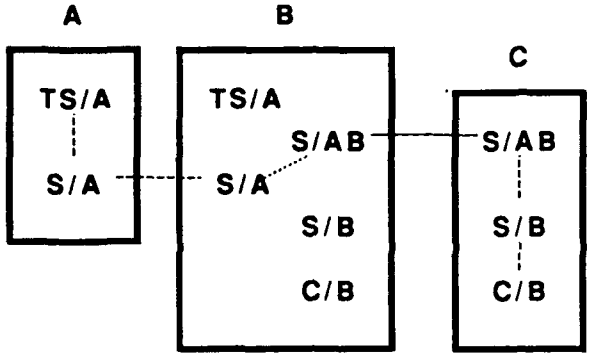


Figure 4. Counterexample for Nesting, Levels Not Totally Ordered

It is easily checked that the accreditation ranges shown for subsystems *A*, *B*, and *C* are convex. Furthermore, they satisfy the original nesting condition. The ranges of *A* and *C* are nested in that of *B*, and the ranges of *A* and *C* are disjoint with one another. A cascading problem exists because the penetration resistance of the cascading path (along the dotted line) is less than the risk index of the range from TS/A to C/B. The risk indexes are defined in this example using the graph in Figure 5.

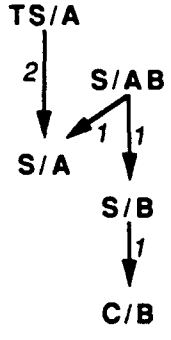


Figure 5. Risk Index Graph

To determine the risk index $R(a,b)$ of a downgrade from a to b, find a on the graph in Figure 5, walk along the graph to b (either with or against the direction of the arrows, as needed), and add up the numbers on the steps taken with the arrows (not counting the steps taken against the arrows). For example, R(TS/A, S/B) = 3. One can check that this risk index assignment is skew monotonic.

The risk index of the range from TS/A to C/B is 4. To find the penetration resistance of the path, note first that the step from S/A to S/AB in subsystem *B* is an upgrade, not a downgrade, so subsystem *B* does not have to be defeated by the attacker. The penetration resistance of the path is therefore only the maximum from subsystems *A* and *C*. Assuming that the penetration resistance of each system is no greater than the risk index of its accreditation range, we see that systems *A* and *C* each have a penetration resistance of only 2.

We could have left out the level C/B from subsystems *B* and *C* and still have had a counterexample. The reason for including them is that, with them, the example also works under any reasonable interpretation of the DOD risk index. The range from TS/A to C/B, if it were supported in a single system, would have a "Yellow Book" risk index greater than the risk index from TS/A to S/A or that from S/AB to C/B.

Top-Adjusted Ranges

Walker's condition, that all the accreditation ranges in the network have the same top level, is sufficient to prevent cascading even when sensitivity levels are partially ordered, as long as accreditation ranges are convex. For, consider the last downgrading step in a cascading path. It occurs in a subsystem whose accreditation range reaches from the maximum sensitivity level in the network all the way down to the least sensitivity level on the path, so this subsystem provides a sufficient penetration resistance.

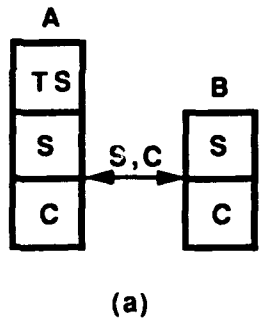
THE MATRIX APPROACH

The Resistance Graph of a Network

The matrix approach begins by representing the network as a directed graph. If we define the "cost" of a path to be its penetration resistance, a standard shortest (i.e., least-cost) path matrix algorithm will then find candidates for cascading paths. A discussion of the applicability of matrix algorithms for different sorts of path cost calculations is given in [6].

The resistance graph is the same graph specified in Appendix C of the TNI to state the cascade condition. The vertices of the graph are not the subsystems of the network, but rather abstract nodes called *protection domains* ("regions" in the TNI) corresponding to the sensitivity levels in the accreditation range of each subsystem. Edges of the graph are either network links between vertices at the same level, or

junctions between different protection domains in the same subsystem. Figure 6 shows a network and Figure 7 its corresponding resistance graph.



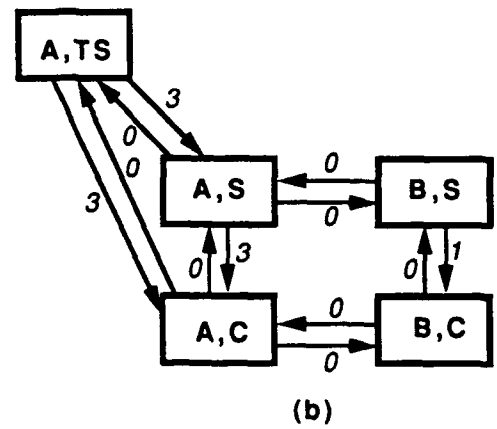


Figure 6. A Network

Figure 7. Resistance Graph

Each vertex is labelled with a pair of the form (s,x) where s is a subsystem and x is a sensitivity label. There are edges between every pair of protection domains in a subsystem, even those whose sensitivity levels are not adjacent.

Edges are labelled with a cost representing their penetration resistance. The cost of an edge representing an illegal flow (i.e., a downgrade) is the penetration resistance of the subsystem. The cost of an upgrade edge within a subsystem, or a network link between subsystems, is zero, since information flows along those edges are legal. Thus, the cost of an edge from an arbitrary domain (s,x) to any other domain (t,y) is:

$$C((s,x),(t,y)) = \begin{cases} 0 \text{ if } x \leq y \\ P(s) \text{ otherwise} \end{cases}$$

Note that one-way network links can be represented in the graph simply by omitting the edge in the disallowed reverse direction.

Cascading Paths

A cascading path in the graph is a directed path whose resistance is less than the risk index of the range it spans. The resistance of a path is the maximum of the cost labels on its edges. This figure is compared with the path risk, i.e., the risk index of a

downgrade from the sensitivity level at the beginning of the path to the sensitivity level at the end of the path.

As remarked earlier, taking the maximum of the resistances along a path is a worst-case or conservative estimate of the penetration resistance of the collection of systems on the path. Other cost computations might be appropriate, in particular network environments, though more complex functions might not be amenable to the matrix treatment.

End-to-End Encryption Links

Ordinarily, a network-link edge in the graph corresponds to a physical link in the usual network sense: a communication channel that does not pass through any intermediate subsystems. There is one important exception. If one subsystem communicates with another through end-to-end encryption devices, the explicit inclusion of those devices as subsystems can mask a cascading problem, unless certain logical links are added. Consider the network in Figure 8.

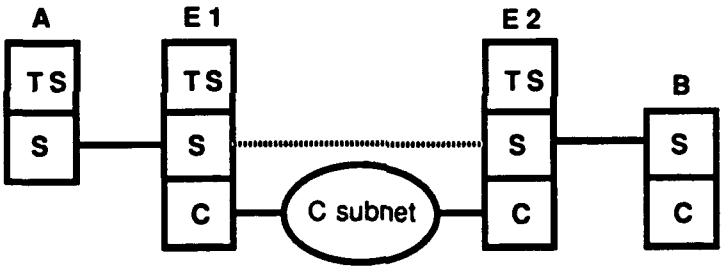


Figure 8. Implicit Link With End-to-End Encryption

In Figure 8, E1 and E2 are end-to-end encryption devices. They are capable of encrypting S-level information in such a way that the ciphertext can be transmitted at the C level. (Ignore the dotted line for the moment.) Their very high penetration resistance is indicated by giving them an accreditation range that spans all levels in the network. But, despite the high assurance of the encryption, this network has a cascading problem.

The cascading path begins at TS in *A* and ends at C in *B*. It does not appear to be a cascading path, because it appears that an attacker would have to defeat system *E1* in order to compromise information along the path. In other words, if the network graph were constructed using only physical links, the analysis would fail to detect the cascading problem.

In actuality, S-level information is passed along from *E1* to *E2* as indicated by the dotted line. There is a cascading problem, because neither *E1* nor *E2* has to be defeated, and an attacker could cascade information from TS to C by defeating *A* and *B*, each accredited only for less.

For this reason, implicit links should be shown between end-to-end encryption devices at each level and subsystem where connections can be established. This would cause the dotted line in Figure 8 to be added to the network graph as an edge (in both directions). Alternatively, one could omit the end-to-end encryption devices entirely, and place edges directly between protection domains that communicate over an end-to-end encrypted connection - in this example, between (*A*,S) and (*B*,S).

A Matrix Algorithm

The resistance graph is represented in matrix form by arbitrarily numbering the vertices $v_1, ..., v_n$. The ij^{th} element of the initial penetration resistance matrix P^0 is the cost of the edge from vertex i to vertex j, if an edge exists, or infinity (∞), if no edge exists. Thus, if E is the set of edges of the resistance graph, we have:

$$P_{ij}^0 = \begin{cases} C(v_i,v_j) \text{ if } (v_i,v_j) \in E \\ 0 \text{ if } i = j \\ \infty \text{ otherwise} \end{cases}$$

The next step is to compute the path resistance matrix P^n, where n is the number of vertices in the graph, i.e., the number of protection domains in the network. The ij^{th} element of P^n is the cost of the minimum-cost path from v_i to v_j. The matrix is calculated recursively using a shortest-path algorithm.

The intermediate matrices P^k, for $k = 1, ..., n - 1$, have a different meaning depending on which matrix algorithm is used. In [10], the k^{th} matrix showed costs of paths of length k or less; in the Floyd-Warshall algorithm given below and in [6], the k^{th} matrix shows costs of paths using only the first k vertices as intermediate steps in the path. The advantage of the Floyd-Warshall algorithm is that the computation takes $O(n^3)$ steps instead of $O(n^3 \log_2 n)$ steps.

The recursion formula for the k^{th} matrix is the following:

$$P_{ij}^k = \text{Min } \{P_{ij}^{k-1}, \text{Max } \{P_{ik}^{k-1}, P_{kj}^{k-1}\}\}.$$

The idea is that to find the shortest path from v_i to v_j using the first k vertices, the answer will be either the path already found using only the first k - 1 vertices, or else a new path with v_k in it. The new path is the concatenation of paths from v_i to v_k and from v_k to v_j, both using only the first k - 1 vertices. (Vertex v_k will not occur twice, since paths with loops can't be minimal-cost.)

To calculate this with a program, begin by initializing an array `P(0,i,j)` with the edge costs as above, using a number larger than any other edge cost in place of infinity. Then the recursion formula is easily implemented with the nested for-loop:

```
for k := 1 to n do
   for i := 1 to n do
      for j := 1 to n do
         P(k,i,j) := MIN(P(k-1,i,j),MAX(P(k-1,i,k),P(k-1,k,j)));
```

The above program finds the penetration resistance of the minimum-resistance path from v_i to v_j, for all vertex pairs. It could be augmented to keep track of the actual path as well, as was done in ANSSR [13].

Detecting Cascading

Having found the path resistance matrix, it remains to check whether any of the minimum-resistance paths are cascading paths. The resistance of each path is compared with the risk index of the range it spans. Define the risk matrix R by:

$$R_{ij} = R(a_i,a_j)$$

where

$$v_i = (s_i,a_i).$$

Then a cascading path exists from v_i to v_j when the resistance is less than the risk; that is, when:

$$P_{ij}^n < R_{ij}.$$

Example

Consider the network and resistance graph in Figure 7. Number the vertices in the resistance graph as follows:

1. (A,TS), 2. (A,S), 3. (A,C), 4. (B,S), 5. (B,C).

The initial penetration resistance matrix P^0 is given by:

$$P^0 = \begin{bmatrix} 0 & 3 & 3 & \infty & \infty \\ 0 & 0 & 3 & 0 & \infty \\ 0 & 0 & 0 & \infty & 0 \\ \infty & 0 & \infty & 0 & 1 \\ \infty & \infty & 0 & 0 & 0 \end{bmatrix}$$

After iterating five times, the path resistance matrix P^5 turns out to be:

$$P^5 = \begin{bmatrix} 0 & 3 & 3 & 3 & 3 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Now we compare this with the risk matrix R computed using the values $R(\text{TS,S}) = 2$, $R(\text{TS,C}) = 3$, $R(\text{S,C}) = 1$.

$$R = \begin{bmatrix} 0 & 2 & 3 & 2 & 3 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Since P^5 dominates R at every element, we conclude that the network has no cascading problem.

SECTION 3

PROBLEM REDUCTION TECHNIQUES

The matrices used to test for cascading are large if there are many subsystems or large accreditation ranges in the network. We consider the large-network problem first.

ANALYSIS OF LARGE NETWORKS

Cascading analysis of a large network is impractical if it is necessary to consider all of the individual computer systems that belong to the network. This is particularly true in the case of the world-wide internet. It would be pleasant if, once we had determined that an interconnection of subsystems had no cascading problem, to treat the whole interconnection as a single subsystem for purposes of higher-level interconnections. Unfortunately, it is not safe to do so, unless additional assumptions are made about the network subsystem.

Recursion Counterexample

Consider the network in Figure 9. Network *X* is the interconnection of two subsystems *A* and *B*, assumed accredited for the indicated ranges. Network *X* does not have any cascading problem.

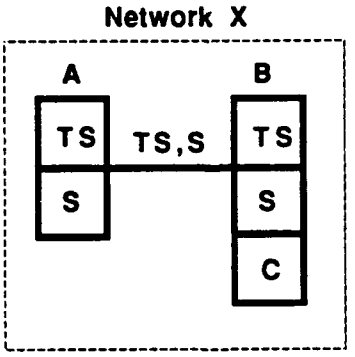


Figure 9. An Interconnection With No Cascading Problem

It is tempting, at this point, to say that network *X* may be considered a single subsystem, accredited for its overall range of TS to C. Suppose we do that, and connect the new subsystem to another accredited system *C*, as shown in Figure 10.

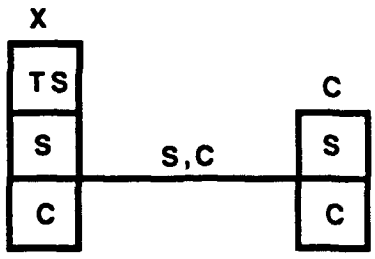


Figure 10. Interconnection of Subsystems *X* and *C*

If we analyze the network in Figure 10 for cascading, the conclusion is that there is no cascading problem, since an attacker would have to defeat subsystem *X*, which is accredited for the full range of the network. But this is false, as we can see by taking a closer look at the network, in Figure 11.

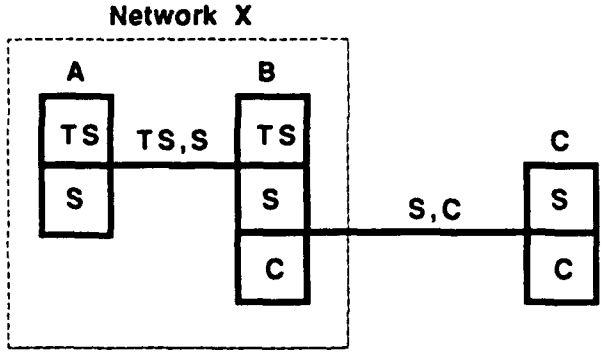


Figure 11. A Closer Look at the Network

In Figure 11, it is easy to see the cascading path:

$$(A,\mathrm{TS}),(A,\mathrm{S}),(B,\mathrm{S}),(C,\mathrm{S}),(C,\mathrm{C}).$$

The penetration resistance of system *B* does not help, since there was no downgrade there. The path resistance is the maximum of those of systems *A* and *C*, and neither is enough for the full range TS to C. Evidently it is not, in general, legitimate to treat the cascade-free interconnection of subsystems as a single subsystem accredited for the full range of sensitivity levels. The example also shows that it does not help to assume that the external connections are made only through maximum-resistance hosts.

A Safe-Recursion Condition

There is, however, a condition that enables us to treat interconnections as single subsystems in a higher-level cascading analysis. It is sufficient to require every network subsystem to have the following property: its multilevel component

subsystems must each have a penetration resistance adequate for the whole network subsystem accreditation range. We could paraphrase this as saying that the network subsystem exhibits a "common level of trust throughout," the way the TNI describes single trusted systems.

PROPOSITION: *Let N be a collection of accredited subsystems. Let A be the smallest accreditation range including each of the accreditation ranges of the subsystems in N. Assume that each multilevel subsystem in N has a penetration resistance P at least equal to the risk index of A. Then N may safely be considered a single accredited subsystem with accreditation range A and penetration resistance P for purposes of cascading analysis.*

To prove this, we consider an interconnection in which some components are network subsystems. We will compare two cascading analyses of this interconnection. First, the *host-level* analysis considers only individual hosts as subsystems. Second, the *network-level* analysis treats network subsystems satisfying the hypothesis as single subsystems. We will show that any cascading path appearing in the host-level analysis will also appear as a cascading path in the network-level analysis.

If the interconnection has a cascading path through the individual hosts, consider each subsystem within which a downgrade has occurred. If that subsystem is a network, a downgrade must have occurred inside one of its component hosts, which by hypothesis has the same penetration resistance as the whole network subsystem. This means that the network subsystem did not contribute any more to the path resistance in the network-level analysis than the host did in the host-level analysis. The same is true for every subsystem in which a downgrade occurs on the path. Consequently, if the path is detected as a cascading path at the host level (because of inadequate resistance), it must also be detected as a cascading path at the network level.

Community Separation

Some hosts may appear to be connected into a huge internet, but in fact are logically isolated into a community of interest using end-to-end encryption. A collection of subsystems that may communicate only among themselves may be considered as a complete network for purposes of cascading analysis. As remarked above where the resistance graph was defined, end-to-end-encrypted channels may be represented as single links, and the intermediate subsystems that handle only encrypted information would not be included in the graph.

REDUCTION OF ACCREDITATION RANGES

Even when a network has only a few subsystems, a large resistance matrix can be generated if their accreditation ranges include a great many sensitivity levels, so that there are many protection domains. This would be the case in an environment supporting a large number of categories or compartments of information. It is often sufficient to include just a few representative protection domains to capture all the essentially different paths.

Starting with a complete resistance graph, one can identify certain vertices as equivalent if they have edges with the same cost to the same or equivalent vertices. Clearly, for any cascading path through one of these vertices, there is another path with the same resistance through any equivalent vertex. Hence, it is only necessary to include one representative from each equivalence class.

A detailed consideration of algorithms for identifying equivalent protection domains and thus reducing the resistance graph is beyond the scope of this paper. The problem is very much like the reduction of finite-state automata. A practical approach would probably begin by identifying likely candidates for equivalence, such as protection domains in the same subsystem whose sensitivity levels differ only by the substitution of one category for another.

SECTION 4

SUMMARY

Cascading is an example of a global risk created by connecting subsystems into a network. It is a potential concern when the range of sensitivity levels spanned by the network is greater than the accreditation range of its multilevel component subsystems. Whether it is actually a problem in a particular network depends partly on the difficult-to-analyze risk of multiple cooperating attacks on different subsystems, and partly on some fairly straightforward considerations based on accreditation ranges and network connectivity.

The first reason for performing a cascading analysis is to see if the concern can be eliminated by showing that cascading is impossible in the given network. If the problem cannot be dismissed, then cascading analysis can focus attention on those subsystems and connections involved in a cascading path, so that more detailed arguments or specific countermeasures may be applied.

Cascading analysis requires a measure of risk based on accreditation ranges, and a corresponding measure of penetration resistance. We have assumed that accreditation ranges are convex and that the risk index function is skew-symmetric. We have also adopted a conservative estimate for the combined penetration resistance of a collection of systems - their maximum.

The two most useful methods to detect cascading are the nesting condition and a matrix approach. The nesting condition that appears in the TNI is sufficient to dismiss cascading, when it is satisfied, in an environment where sensitivity levels are linearly ordered. We have exhibited an example showing that the original nesting condition is not sufficient in a case where sensitivity levels are partially but not linearly ordered, though the generalized nesting condition is known to be sufficient.

The matrix approach involves constructing the resistance graph of a network. Its vertices are protection domains, and its edges represent possible information transfers whose costs depend on subsystem penetration resistance. Least-resistance paths between all pairs of vertices are calculated at once using a matrix shortest-path algorithm, which is easily adapted to keep track of the actual least-cost paths. Cascading paths are those whose resistance is less than their risk index.

Large networks present a computational problem which cannot always be handled by treating component networks as single accredited systems, as we have shown by counterexample. A sufficient condition to allow a network to be treated as a single subsystem was given: its multilevel subsystems must have sufficient penetration resistance for the network range. Sometimes a network is not as large as it looks, because parts of it are isolated cryptographically. When the size of the resistance graph is due to a large number of sensitivity levels in accreditation ranges, we observe that equivalence classes can be formed to reduce the matrix. The effectiveness of computational techniques for doing so would be a good topic for future investigation.

LIST OF REFERENCES

1. Department of Defense Directive 5200.28, "Security Requirements for Automated Information Systems (AISs)," March 21, 1988.

2. National Computer Security Center, "Trusted Network Interpretation of the Trusted Computer System Evaluation Criteria," NCSC-TG-005, 31 July 1987.

3. "Proceedings of the Department of Defense Computer Security Center Invitational Workshop on Network Security," New Orleans, LA, March, 1985.

4. National Computer Security Center, "Trusted Network Interpretation Environments Guideline," NCSC-TG-011, 1 August 1990.

5. E. J. Powanda and J. W. Genovese, "Configuring a Trusted System Using the TNI," *Fourth Aerospace Computer Security Applications Conference*," IEEE Catalog No. 88CH2619-5, December, 1988, pp. 256-261.

6. J. A. Fitch, III, and L. J. Hoffman, "A Network Shortest Path Security Model," GWU-IIST-90-32, The George Washington University, September, 1990.

7. T. M. P. Lee, "Statistical Models of Trust: TCBs vs. People," Proc. 1989 IEEE Symposium on Security and Privacy, pp. 10-19.

8. DOD Computer Security Center, "Computer Security Requirements - Guidance for Applying the Department of Defense Trusted Computer System Evaluation Criteria in Specific Environments," CSC-STD-003-85, 25 June 1985.

9. "Department of Defense Trusted Computer System Evaluation Criteria," DOD 5200.28-STD, December, 1985.

10. J. K. Millen and M. W. Schwartz, "The Cascading Problem for Interconnected Networks," *Fourth Aerospace Computer Security Applications Conference*," IEEE Catalog No. 88CH2619-5, December, 1988, pp. 269-274.

11. J. K. Millen, "Algorithm for the Cascading Problem," in J. P. Anderson (editor), *Internet IEEE Cipher News Group*, June, 1990.

12. J. K. Millen, "Interconnection of Accredited Systems," *Third Aerospace Computer Security Applications Conference*," AIAA, December, 1987, pp. 60-65.

13. D. J. Bodeau, F. N. Chase, and S. G. Kass, "ANSSR: A Tool for Risk Analysis of Networked Systems," *13th National Computer Security Conference*, National Institute of Standards and Technology and National Computer Security Center, October, 1990, pp. 687-696.